



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/882,172	06/14/2001	John Mark Hartel	AUS920010225US1	7996
35525	7590	02/02/2009	EXAMINER	
IBM CORP (YA)			KE, PENG	
C/O YEE & ASSOCIATES PC				
P.O. BOX 802333			ART UNIT	
DALLAS, TX 75380			PAPER NUMBER	
			2174	
			NOTIFICATION DATE	
			DELIVERY MODE	
			02/02/2009	
			ELECTRONIC	

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ptonotifs@yeeiplaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte JOHN MARK HARTEL, JOHN CONRAD SANCHEZ, and
KARALEE BROWN LEBLANCE

Appeal 2008-0801
Application 09/882,172
Technology Center 2100

Decided: [Date of mailing]

Before HOWARD B. BLANKENSHIP, JEAN R. HOMERE, and
CAROLYN D. THOMAS, *Administrative Patent Judges*.

BLANKENSHIP, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

This is an appeal under 35 U.S.C. § 134(a) from the Examiner's final rejection of claims 1-37 and 39-41, which are all of the claims remaining in this application. We have jurisdiction under 35 U.S.C. § 6(b).

We affirm the Examiner's rejections against claims 1-37 and 39-41.
We add a new ground of rejection under 35 U.S.C. § 101 against claims
25-36.

Invention

Appellants' invention relates to a method in a data processing system for editing a property. The method identifies one or more methods invoked by a property editor, selects a graphical user interface based on the one or more invoked methods, and provides the graphical user interface for use in editing the property.

Representative Claim

1. A method, in a data processing system, for editing a property, comprising:

identifying one or more methods invoked by a property editor associated with the property;

selecting a graphical user interface based on the one or more methods invoked by the property editor; and

providing the graphical user interface for use in editing the property.

Prior Art

The Examiner relies on the following references:

Lindhorst	US 6,337,696 B1	Jan. 8, 2002
Zimmerman	US 6,417,872 B2	Jul. 9, 2002
Miller	US 6,661,437 B1	Dec. 9, 2003

Examiner's Rejections

I. Claims 1-3, 7, 8, 13-15, 19, 20, 25-27, 31, 32, and 37 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Miller.

II. Claims 4-6, 16-18, and 28-30 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Miller and Lindhorst.

III. Claims 9, 12, 21, 24, 33, 36, 40, and 41 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Miller and Zimmerman.

IV. Claims 10, 11, 22, 23, 34, 35, and 39 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Miller, Zimmerman, and Lindhorst.

Claim Groupings

Based on Appellants' arguments in the Appeal Brief, we will decide the appeal on the basis of claims 1, 3, 4, 7, 9, 10, 40, and 41. *See* 37 C.F.R. § 41.37(c)(1)(vii).

FINDINGS OF FACT

1. Miller discloses selecting a "subtitle language" menu item (Fig. 5, element 545).

2. Miller's selecting identifies and invokes methods of editing a subtitle language property by a property editor of the subtitle language property. Specifically, the subtitle language property is edited by invoking a method of changing the subtitles from one language to another (Fig. 5 and col. 6, l. 65 - col. 7, l. 18).

3. Based on the invoked method, a graphical user interface such as a dialog box is selected for editing the subtitle language property (Fig. 5, element 555).

4. The graphical user interface is provided to the user for editing the subtitle language property (Fig. 5 and col. 6, l. 65 - col. 7, l. 18).

5. The graphical user interface is selected based on editing methods that are identified when the user selects the appropriate menu item (Fig. 5 and col. 6, l. 65 - col. 7, l. 18).

6. Appellants' Specification states: "A 'property' as the term is used herein, refers to anything that can be represented by a Java class, i.e. a data type. Examples of properties include text color, background color, text string, windspeed, percentage, computer operating system, and the like." (Spec. 14: 4-8).

7. Appellants' Specification discloses embodiments of a Java PropertyEditor Interface (e.g., Spec. 14: 14 - 17: 15), but we do not find any limiting definition for the term "PropertyEditor Interface."

8. Miller discloses "an on-screen User Interface display . . . for maintaining User viewability of the currently selectable menu items" (col. 6, l. 65 to col. 7, l. 1).

9. Appellants' Specification describes embodiments of a custom property editor (e.g., Spec 14: 9 - 15: 7).

10. Zimmerman discloses a "GetPredefinedStrings" function that returns character strings such as red, blue, and green (col. 8, ll. 55-67).

11. Zimmerman discloses the MapPropertyToPage function enables switching from a per-property browsing list to a property sheet page containing the particular property (col. 8, ll. 20-23).

12. Appellants' Specification indicates that the present invention may be embodied in transmission-type "computer readable media," such as digital and analog communications links. (*See Spec. 29: 3-15*).

PRINCIPLES OF LAW

Claim Interpretation

The *claims* measure the invention. *See SRI Int'l v. Matsushita Elec. Corp.*, 775 F.2d 1107, 1121 (Fed. Cir. 1985) (en banc). During prosecution before the USPTO, claims are to be given their broadest reasonable interpretation, and the scope of a claim cannot be narrowed by reading disclosed limitations into the claim. *See In re Morris*, 127 F.3d 1048, 1054 (Fed. Cir. 1997); *In re Zletz*, 893 F.2d 319, 321 (Fed. Cir. 1989); *In re Prater*, 415 F.2d 1393, 1404-05 (CCPA 1969).

"Giving claims their broadest reasonable construction 'serves the public interest by reducing the possibility that claims, finally allowed, will be given broader scope than is justified.'" *In re American Academy of Science Tech Center*, 367 F.3d 1359, 1364 (Fed. Cir. 2004) (quoting *In re Yamamoto*, 740 F.2d 1569, 1571 (Fed. Cir. 1984)). "An essential purpose of patent examination is to fashion claims that are precise, clear, correct, and unambiguous. Only in this way can uncertainties of claim scope be removed, as much as possible, during the administrative process." *Zletz*, 893 F.2d at 322. "Construing claims broadly during prosecution is not unfair to

the applicant . . . because the applicant has the opportunity to amend the claims to obtain more precise claim coverage.” *American Academy*, 367 F.3d at 1364.

Our reviewing court has repeatedly warned against confining the claims to specific embodiments described in the specification. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1323 (Fed. Cir. 2005) (en banc). “Though understanding the claim language may be aided by explanations contained in the written description, it is important not to import into a claim limitations that are not part of the claim. For example, a particular embodiment appearing in the written description may not be read into a claim when the claim language is broader than the embodiment.” *Superguide Corp. v. DirecTV Enterprises, Inc.*, 358 F.3d 870, 875 (Fed Cir. 2004) (citing *Electro Med. Sys. S.A. v. Cooper Life Sci., Inc.*, 34 F.3d 1048, 1054 (Fed. Cir. 1994)).

While the patent specification includes an embodiment showing the use of “a C++ class hierarchy derivation of the program logic,” . . . and uses terms characteristic of object-oriented programming in connection with that example, neither the written description nor the claims anywhere state or imply that the invention must use object-oriented programming in general, or C++ in particular. Without more, the use of an example that employs object-oriented programming is not sufficient to require that the claims be limited to embodiments using C++ or a similar programming language.

Tivo, Inc. v. EchoStar Communications Corp., 516 F.3d 1290, 1307 (Fed. Cir. 2008).

Anticipation

“Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim.” *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 730 F.2d 1452, 1458 (Fed. Cir. 1984).

For a prior art reference to anticipate in terms of 35 U.S.C. § 102, every element of the claimed invention must be identically shown in a single reference. However, this is not an “ipsissimis verbis” test. *In re Bond*, 910 F.2d 831, 832 (Fed. Cir. 1990).

Obviousness

The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art, (2) any differences between the claimed subject matter and the prior art, and (3) the level of skill in the art. *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966).

What a reference teaches is a question of fact. *In re Baird*, 16 F.3d 380, 382 (Fed. Cir. 1994); *In re Beattie*, 974 F.2d 1309, 1311 (Fed. Cir. 1992). The presence or absence of a motivation to combine references in an obviousness determination is a pure question of fact. *In re Gartside*, 203 F.3d 1305, 1316 (Fed. Cir. 2000).

Statutory Subject Matter

“A transitory, propagating signal is not a ‘process, machine, manufacture, or composition of matter.’ Those four categories define the

explicit scope and reach of subject matter patentable under 35 U.S.C. § 101.” *In re Nuijten*, 500 F.3d 1346, 1357 (Fed. Cir. 2007). “If a claim covers material not found in any of the four statutory categories, that claim falls outside the plainly expressed scope of § 101 even if the subject matter is otherwise new and useful.” *Id.* at 1354.

ANALYSIS

I. Section 102 rejection of claims 1, 2, 13, 14, 25, and 26

Appellants argue that Miller does not disclose selecting a graphical user interface based on the one or more methods invoked by a property editor associated with a property, because only one graphical user interface is selected to edit a given property (Br. 14). Appellants further submit that in Miller the user interface components for a parameter are static, and the same graphical user interface will always be provided for editing a property when a user selects the appropriate menu item (*id.*).

The Examiner found that “Miller teaches this limitation because each option on its menu is a method for a property. Appellant has defined a ‘property’ as a data type, which includes text color, background color, text string, windspeed, percentage, computer operating system, and the like.” (Ans. 12) (citations omitted).

Issue: Have Appellants shown that the Examiner erred in finding that Miller discloses selecting a graphical user interface based on the one or more methods invoked by the property editor?

Miller describes identifying and selecting a subtitle language property using an editor associated with the property (FF 1-2). Miller further

describes selecting a graphical user interface dialog box based on the one or more methods invoked by the property editor (FF 3) and providing the graphical user interface for use in editing the property (FF 4-5). Thus, we find that Miller's selecting the user interface for editing a subtitle language property is consistent with selecting a graphical user interface based on methods invoked by the property editor.

Appellants further argue that the terms "method," "property," and "property editor" should be limited to object-oriented runtime environments such as a Java runtime environment as disclosed in the Specification (Br. 15-16).

Issue: Have Appellants shown that the claims are limited to object-oriented runtime environments?

Although the claim language may be aided by explanations contained in the written description, Appellants cannot import into a claim limitations that are not part of the claim. Here, Appellants are trying to improperly read particular embodiments appearing in the written description into a claim when the claim language is broader than the embodiment. In view of Appellants' broad definition of "property editor" in the Specification (FF 6), and absent an improper reading of embodiments from Appellants' written description into the terms of representative claim 1, we are not persuaded that the claims are limited to object-oriented runtime environments. Further, Appellants have provided no evidence in support of the view that the terms "method," "property," or "property editor" have ordinary and customary meanings in the pertinent art that would distinguish over Miller's disclosure. Thus, Appellants have not shown that Appellants' Specification or the

claims state or imply that the invention must use object-oriented runtime environments in general, or Java in particular.

Therefore, we sustain the rejection of claims 1, 2, 13, 14, 25, and 26 under 35 U.S.C. § 102(e) as being anticipated by Miller.

II. Section 102 rejection of claims 3, 15, 27 and 37

Appellants argue that the claim 3 recitation “PropertyEditor Interface” is defined in the Specification, and limited to that definition (Br. 16).

The Examiner found that “by allowing [the] user to adjust the properties of a device through an interface, Miller teaches a property editor.” (Ans. 13).

Issue: Have Appellants shown that the term “PropertyEditor Interface” is limited by Appellants’ Specification?

Appellants’ Specification merely discloses embodiments of a Java PropertyEditor Interface (FF 7). However, Appellants have not shown and we do not readily find a lexicographic definition for the term “PropertyEditor Interface” in the Specification. The scope of a claim cannot be narrowed by reading disclosed embodiments into the claim. The claim language itself measures the invention. Absent an improper reading of Appellants’ “PropertyEditor Interface” methods from the written description into representative claim 3, we are not persuaded of error in the Examiner’s findings.

Therefore, we sustain the rejection of claims 3, 15, 27, and 37 under 35 U.S.C. § 102(e) as being anticipated by Miller.

III. Section 102 rejection of claims 7, 8, 19, 20, 31, and 32

Appellants argue that Miller does not teach or suggest that an ability of the property editor includes “an ability to edit a property using tags, and the graphical user interface includes at least one of a popup choice selection area virtual button and a current selection display field” as recited in claim 7 (Br. 17).

The Examiner found that “Miller teaches this limitation because the menu for subtitle has a popup choice selection that includes a currently selected field.” (Ans. 14).

Issue: Have Appellants shown that the Examiner erred in finding that Miller discloses that the graphical user interface includes at least one of a popup choice selection area virtual button and a current selection display field?

Miller discloses an on-screen display that displays currently selectable items (FF 8). Here, Miller discloses selection tags as part of the graphical user interface. These tags represent a current selection display field because once a tag is selected, the tag displays the user’s current selection of the subtitle language property. Thus, we find that claim 7’s “current selection display field” reads on Miller’s on-screen display selection tags.

Therefore, we are not persuaded of error in the Examiner’s finding of anticipation with respect to representative claim 7.

We sustain the rejection of claims 7, 8, 19, 20, 31, and 32 under 35 U.S.C. § 102(e) as being anticipated by Miller.

IV. Section 103 rejection of claims 4-6, 16-18, and 28-30

Appellants argue, in defense of claim 4, that there is no motivation in the prior art for combining Miller and Lindhorst, because Miller makes no mention of an object-oriented runtime environment or editing of event handlers (Br. 18-19).

The Examiner found:

Miller's menu options are designed for personal computer and other Internet compatible device. Lindhorst teaches a software program for creating and editing event handlers that are linked to an object with HTML and other Internet documents. Therefore, in order for an Internet preference menu to operate on a computer or an internet compatible device, a programmer would want to combine Lindhorst's teaching with Miller's method.

(Ans. 14) (citations omitted).

Issue: Have Appellants shown that the claims are limited to object-oriented runtime environments which affect the motivation to combine the references?

As noted *supra* regarding claim 1, we do not interpret Appellants' claims as being limited to object-oriented runtime environments. Furthermore, Appellants had the opportunity to amend the claims to be commensurate with the arguments made. However, in this case the claims are drafted far broader than the arguments relied upon. The Examiner has articulated a reason for combining the references and Appellants' reliance on the claims being limited to object-oriented runtime environments to dispute such a motivation is flawed.

Therefore, we are not persuaded of error in the Examiner's rejection of representative claim 4, and thus sustain the rejection of claims 4-6, 16-18, and 28-30 under 35 U.S.C. § 103(a) as being unpatentable over Miller and Lindhorst.

V. Section 103 rejection of claims 9, 12, 21, 24, 33 and 36

Appellants allege, in defense of claim 9, that the Specification provides an "exemplary definition" of a custom editor as an editor that uses methods created by a programmer rather than defined by the programming language, thus distinguishing over Zimmerman (Br. 22-23).

However we find that Appellants have not shown and we do not readily find any limiting definition in the Specification for the term "custom editor" (FF 9).

Appellants further allege (for claim 9) that Miller and Zimmerman cannot be properly combined, because Miller does not recognize a problem for which the teachings of Zimmerman would be a solution (Br. 24). Appellants also argue that Miller is non-analogous art because Miller is not directed towards property editors in an object-oriented runtime environment (App. Br. 23).

As noted *supra*, "object-oriented runtime environment" is not in the claims and there is no basis for reading this limitation into representative claim 9. Further, Appellants neglect to explain how alleging that a reference's (Miller's) alleged lack of recognizing a problem for which the teachings of another reference (Zimmerman) would be a solution demonstrates a defect in a prima facie case for obviousness, or might

otherwise tend to show nonobviousness. Appellants have not addressed the Examiner's finding that the artisan would have incorporated the method of Zimmerman with the method of Miller to provide a single application having only a desired functionality to users without requiring creation of several versions of the application (Ans. 7).

Therefore, we are not persuaded of error in the Examiner's rejection. We sustain the § 103(a) rejection of claims 9, 12, 21, 24, 33, and 36 over Miller and Zimmerman.

VI. Section 103 rejection of claim 40

Appellants argue, in defense of claim 40, that the getTags method in Table 1 of Appellants' Specification returns an array of tags if the property value must be one of a set of known tagged values (Br. 24).

The Examiner finds that Zimmerman discloses a getTags method (Ans. 8).

Issue: Have Appellants shown that the Examiner erred in finding that Zimmerman discloses a getTags method?

Zimmerman discloses a function that returns an array of character strings (FF 10). Appellants have failed to show how Zimmerman's return of such an array of tags is distinguishable from the claimed "getTags." Absent an improper reading of details of the "getTags" method described in Appellants' written description into claim 40, we are not persuaded of error in the rejection.

Therefore, we sustain the § 103(a) rejection of claim 40 over Miller and Zimmerman.

VII. Section 103 rejection of claim 41

Appellants argue, in defense of claim 41, that the MapPropertyToPage function is not the same as the getCustomEditor method described in Table 1 of Appellants' Specification (Br. 24-25).

The Examiner finds that the MapPropertyToPage function in Zimmerman is synonymous with the getCustomEditor method (Ans. 9).

Issue: Have Appellants shown that the Examiner erred in associating Zimmerman's MapPropertyToPage function with the getCustomEditor method?

Here, Appellants are again trying to improperly read particular embodiments appearing in the written description into a claim when the claim language is broader than the embodiment. Appellants have not shown how the claim language distinguishes the getCustomEditor function from Zimmerman's MapPropertyToPage function. Absent an improper reading of details of Appellants' disclosed "getCustomEditor" methods into claim 41, we are not persuaded of error in the Examiner's rejection.

Therefore, we sustain the § 103(a) rejection of claim 41 over Miller and Zimmerman.

VIII. Section 103 rejection of claims 10, 11, 22, 23, 34, 35 and 39

Appellants contend that "neither *Miller*, *Zimmerman*, nor *Lindhorst* teaches or suggests an ability to edit a property using a custom editor interface" (App. Br. 25).

The Examiner found that Miller, Zimmerman and Lindhorst teach the above noted feature and that Lindhorst teaches a graphical user interface that includes at least one of a text entry field and an entry error indicator (Ans. 9).

Issue: Have Appellants shown that the Examiner erred in finding that Miller, Zimmerman, and Lindhorst disclose an ability to edit using a custom editor interface?

For the most part, Appellants have not addressed the Examiner's finding that the artisan would have combined the teachings of Miller and Lindhorst to allow a user to custom edit properties that must be input as strings. Appellants merely argue that neither reference teaches or suggests the above-noted limitations without providing any meaningful analysis that explains why the Examiner erred.

Therefore, we are not persuaded of error in the Examiner's rejection and sustain the § 103(a) rejection of claims 10, 11, 22, 23, 34, 35, and 39 over Miller, Zimmerman, and Lindhorst.

IX. New Ground of Rejection -- 37 C.F.R. § 41.50(b)

We reject claims 25-36 under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

Claims 25 through 36 purport a "computer program product in a computer readable medium." As taught by Appellants' Specification, the claimed subject matter may be embodied in signal bearing media, i.e., a digital link, which is not within any statutory class (FF 12). Thereby, the

claim covers material not found in any of the four statutory categories.
Therefore, claims 25-36 fall outside the plainly expressed scope of § 101.

CONCLUSIONS OF LAW

We conclude that Appellants have not shown that the Examiner erred in rejecting claims 1-37 and 39-41.

Thus, claims 1-37 and 39-41 are not patentable.

Since we have entered a new ground of rejection against claims 25-36, our decision is not a final agency action.

DECISION

The Examiner's rejections of claims 1-37 and 39-41 are affirmed.

In a new ground of rejection, we have rejected claims 25-36 under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

With respect to the affirmed rejection(s), 37 C.F.R. § 41.52(a)(1) provides that "Appellant may file a single request for rehearing within two months from the date of the original decision of the Board."

In addition to affirming the Examiner's rejection(s) of one or more claims, this decision contains a new ground of rejection pursuant to 37 C.F.R. § 41.50(b). 37 C.F.R. § 41.50(b) provides that "[a] new ground of rejection pursuant to this paragraph shall not be considered final for judicial review."

37 C.F.R. § 41.50(b) also provides that Appellants, WITHIN TWO MONTHS FROM THE DATE OF THE DECISION, must exercise one of the following two options with respect to the new ground of rejection to avoid termination of the appeal as to the rejected claims:

(1) *Reopen prosecution.* Submit an appropriate amendment of the claims so rejected or new evidence relating to the claims so rejected, or both, and have the matter reconsidered by the examiner, in which event the proceeding will be remanded to the examiner. . . .

(2) *Request rehearing.* Request that the proceeding be reheard under § 41.52 by the Board upon the same record. . . .

Should Appellants elect to prosecute further before the Examiner pursuant to 37 C.F.R. § 41.50(b)(1), in order to preserve the right to seek review under 35 U.S.C. §§ 141 or 145 with respect to the affirmed rejection, the effective date of the affirmance is deferred until conclusion of the prosecution before the Examiner unless, as a mere incident to the limited prosecution, the affirmed rejection is overcome.

If Appellants elect prosecution before the Examiner and this does not result in allowance of the application, abandonment or a second appeal, this case should be returned to the Board of Patent Appeals and Interferences for final action on the affirmed rejection, including any timely request for rehearing thereof.

Appeal 2008-0801
Application 09/882,172

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a).

AFFIRMED
37 C.F. R. § 41.50(b)

pgc

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380